17

A Different Computer Paradigm

Paradigms

The term "paradigm" refers to the over-arching and invisible ideas of our lives.

When everyone agrees about ideas, it is usually an unseen paradigm.

Today, the unseen paradigm of the computer field is called "Computer literacy." It concerns the nature of files, applications, interfaces, and the traditions of file naming.

To question any part of this paradigm is to question all of it.

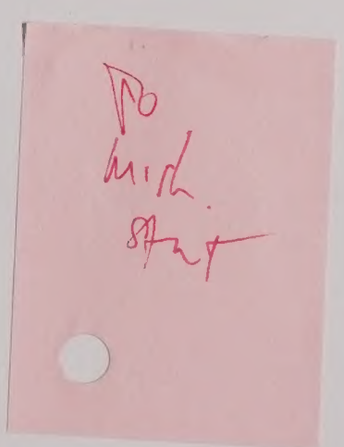
Media

We have many traditions of paper ...

What are media, and what should they become in the digital future? These decisions are being made by competing technocrats in a wild atmosphere. I believe they do not take into account what we need or the nature of human media.

The Xanadu paradigm has always been intended to create a world of electronic documents which
- connect sideways (for commentary, overlay, and natural parallelisms)
- are re-usable on a frictionless basis.

In today's Internet world, that is being redefined around the concepts of transcopyright, transpublishing servers, etc.



Structure

What is structure? What is the true nature of structure?

Many people seem to think that structure is truly hierarchy.

Here are some "fundamental structures" that different computer people believe in. They could even be called computer religions.

- Hierarchical directories and files in them

- Databases
-- Relational database
-- Object-oriented database
(- UMI (from "Booch Method") as a universal methodology of structural creation)

- "Knowledge Representation Systems" (KRS)

- QuickTime as a container for everything (Apple Religion)

- The World Wide Web

To: mj, andrew, ed, !friends
From: Ted Nelson <ted@xanadu.net>
Subject: :jpn: Japanese bullfighting (surprise to me)
Cc: tedx
Bcc:
Attached:

:jpn: Japanese bullfighting (surprise to me)

Hi frenzies--

Well, I've seen it all.

On late-night TV they showed Japanese bullfighting. It exists. I videoed it with my camera from the screen as proof.

It appears to take place way out in the country, with (in the arena shown) about a 180o arc of concrete bleachers.

The trainers-- wearing satin industrial uniforms with baggy pants, team windbreakers and baseball caps, one a jolly leathery thin old man-- lead in their nervous bulls by nose rings. Both contestants look scared. (Animals here often seem to look frightened.)

I didn't see them actually set a match up, but the actual event is like this: the bulls push each other head to head until one gives up. Their overlapping horns do not gore, but kind of guide the continuing pressure. Both bulls look terribly worried throughout the proceedings.

The losing bull definitely looks more demoralized than the winner. One broke and ran in seeming terror.

Then it hit me: it's BULL SUMO! One pushes the other out.

fwIntro
00.01.30 (d7

To Ted Nelson Home Page

FLOATING WORLD™
A NEW USER ENVIRONMENT

NO ICONS • NO METAPHORS • NO FILES • NO APPLICATIONS

Not in Kansas any more

For the last twenty years, as the personal computing world spilled forth, I have been increasingly indignant about all of it.

The sheer impossibility of everything I wanted to do with computers filled me with rage. The clumsy unconnected windows! The tyranny of "applications" owned by different software manufacturers! The clumsy one-step "undo"-- you can undo only one operation, and only in an application, not in the operating system! The prisons of "files" and their naming! (Conventional computer file structures have created a horrible world, and nobody realizes it, nor scarcely perceives the mess of contradictory workarounds they have engendered.) Font hell! Backup hell! Data loss hell! Delirious features that don't work anyway, cretinous dialog and pop-up windows and no-help systems.

The fundamental interface, designed at Xerox PARC in 1974, has not improved since that time, and has become a prison. And one of the tragedies of our time has been "word processing", especially as tangled with clumsy filing, clumsy naming, and the "clipboard"-- the computer guys thought text was simple, and therefore gave the world woefully inadequate tools.

But there is no way to get a foothold without BUYING IN-- without accepting somebody else's definition of the world. The Macintosh, Director, C++, HTML, you name it (even Unix)-- all are clumsy, and in my view, twisted tools that won't do what I think is important.

In tomorrow's computer world, the avalanche of digital media will only increase. We must find a way to tame and manage it, the way e-mail procedures have tamed and managed (to an extent) the flood of email. But with today's naming and filing

conventions, digital media management is appalling difficult and requires far too much attention.

As Archimedes is quoted as saying, "Give me a lever long enough and I will move the world." And, of course, a fulcrum.

What we need is a change at the operating system level.

Where to begin?

THE FULCRUM

I always thought that a system I designed in the 1980s, ZigZag™, had promise. It is a construct universe that lets you build any structure fairly easily, and also see it. It consists of multidimensional list structures with built-in row-and-column visualizations. (If you are not familiar with ZigZag, there is a tutorial.)

It took a long time to get ZigZag implemented, mainly because of the bizarre data structure required (new dimensions may be created at any instant, making conventional arrays impossible). But now it has been prototyped and proven. In 1997, with the help of Andrew Pam of Xanadu Australia, I got a prototype running, and it has proven to be what I had imagined and visualized nearly twenty years ago.

And now, from this unusual and capable mechanism, with the help of Tuomas J. Lukka at Jyväskylä University in Finland, perhaps we can begin again.

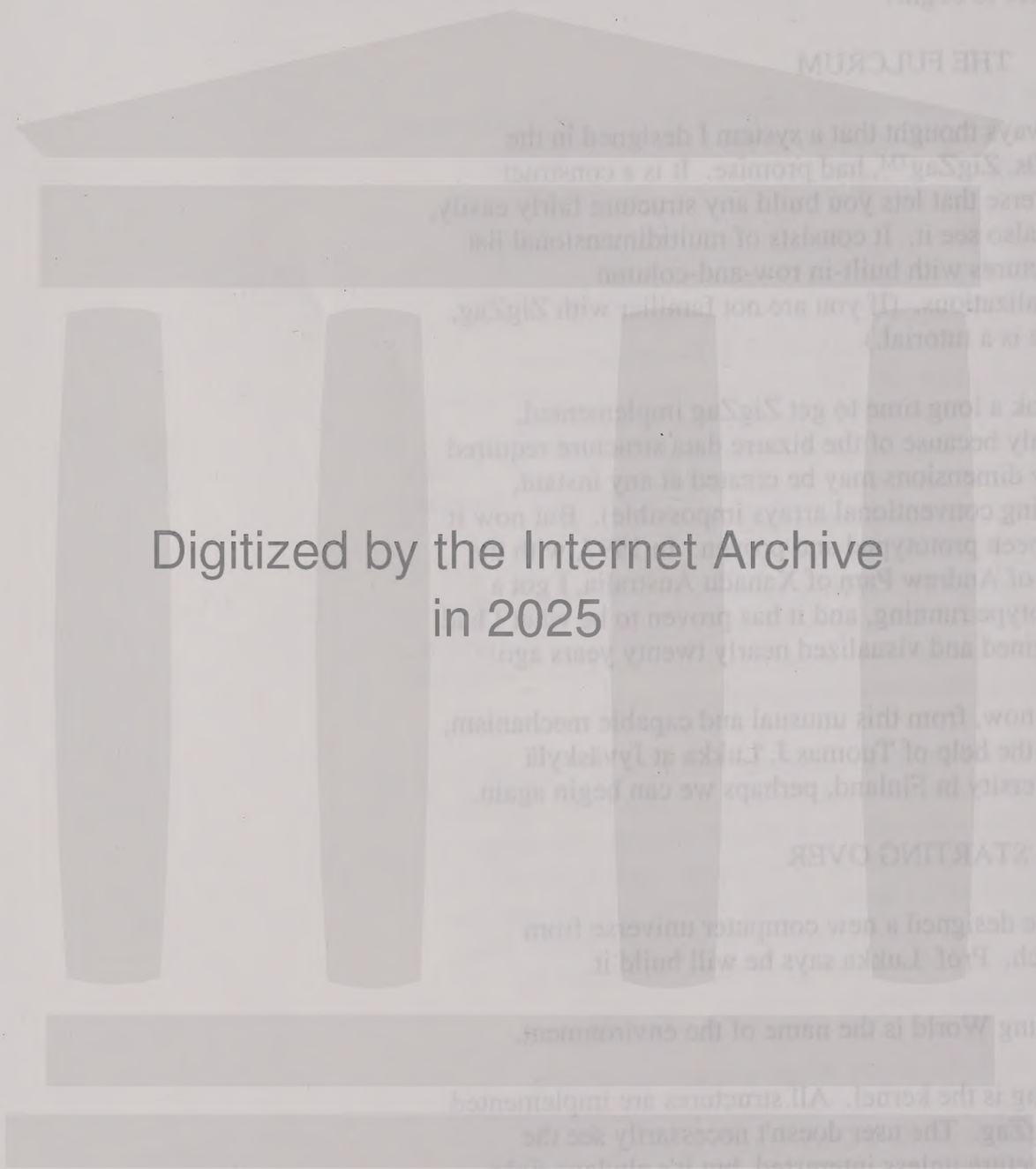
STARTING OVER

I have designed a new computer universe from scratch. Prof. Lukka says he will build it.

Floating World is the name of the environment.

ZigZag is the kernel. All structures are implemented in ZigZag. The user doesn't necessarily see the zzstructure unless interested, but it's alwlays right there and visible if you look under the hood.

Nearly everything in Floating World is composed of ZigZag cells (*zzcells*), ZigZag lists (*zzlists *) and permascrolls (explained below); and though occasionally we will also have to use conventional



Digitized by the Internet Archive
in 2025

https://archive.org/details/differentcompute00tedn

files, for conventional data structures which already are locked in.

ELEMENTS OF FLOATING WORLD

I. STRUCTURES

The fundamental unit is the *thinglet*. It is not an object, an entity, a file, a table, a record, or those other nouns whose computer meanings are already committed. It is something different. Therefore a different name.

A thinglet is composed of zzcells. It has a *maincell*. A maincell is that cell from which the path to any of its other functional cells may be defined. The maincell has a *handle*, free to be listed and arranged in all dimensions except d.handle, along which it attaches to the maincell.

All the internal operations of Floating World are based on thinglets, their handles and maincells, and paths between various significant cells of the thinglets.

The fundamental user-significant unit is the *item*. An item is anything which interests a user, including: • free-floating notes • sticky notes • projects • documents • versions • graphics. Items may have many different user-definable structures, but they are fully reachable through ZigZag visualizations and construction methods.

(RICH TEXT STRUCTURE - AND I DON'T MEAN FONTS)

Text must be simple for the user, but text operations will not be simple internally, and items consisting of text may be quite complex. This is because we afford full Xanadu® connectivity to all text, even titles (see below). We relegate text to being a special case of fluid media (below).

The fundamental visualizable unit of Floating World is the *hyperflob*, or multidimensional flying object. While many hyperflobs (or flobs for short)

may appear on the screen as simple two-dimensional shapes, additional dimensions may be dynamically added at any time. If you want to visualize a hyperflob, think of the old Universal Studios 3D emblem, Saturn-like, with lettering rotating around it-- then morphing into a teapot, pouring text from its spout.

All the visualizations of Floating World are based on hyperflobs and their coordinates, especially as manipulated through *edge lists*. An edge list is a zzlist of elements placed in ascending order along a single coordinate. As an element moves in the space, its coordinates change, and its ordinal position in the edge lists may also change. Tests and interpolations on these edge lists are fundamental FW operations. (This extends easily to multidimensional structures.)

Using these mechanisms, we can model continuous coordinate spaces (*projecton spaces*) with any dimensionality and rules. These spaces may be continuous or discrete, even though the inner structure is discrete zzlists.

We are also designing a user programming language, *CLANG* (cell language), With Clang, we hope to make it possible for the user easily to control the multidimensional behavior of user-designed hyperflobs. It will allow both casual programming (recording keystrokes to create transposable macros) and serious programming, up to and including parallel multiprocessor programs. However, it will of course not compare in efficiency to what skilled programmers do through more conventional tools.

II. FLUID MEDIA

Cells are the structure, contents are not. Contents are either conventional files (like jpg and gif) or *fluid media*. Fluid media are digital contents which are divisible into sequentially addressable elements. Fluid media include text, audio samples, video frames, and faxes.

1

Fluid media are given stable, permanent addresses on *permascrolls* (append-and-read-only files).

2

They are addressed referentially by lists (like the EDL in video editing). As in previous Xanadu designs, change is additive.

3
They may be linked to, and used in different places at the same time (transcluded), to the element level of granularity.

4
Links are based on the Xanadu xu88 model, where each link has two endsets and a type.

5
Fluid media are addressed and connected according to the Xanadu addressing model, managing both transclusions and content links.

6
Transclusions are discovered by address comparison.

III. ZZSTRUX + FLUID MEDIA + XUSTRUX

The combination of zzstructure, hyperflobs, fluid media and Xanadu connectivity allows virtually any presentation and media structure to be created and made radically interactive.

SYNCHRONIZATION

For transmission, cells are also given permanent names and their contents locked. Thus Floating World contents may be exchanged with all contents permanently identified and no confusion among them.

The transmission medium (a mime type) will transport plexes of cells and their associated fluid media. It will be possible to send for parts which are missing. Thus complex structures may be transmitted and synchronized between machines and users without difficulty, and without forcing destruction of previous versions (like most synchronization tools).

IV. DIFFERENCES FROM THE ORDINARY COMPUTER WORLD

Here are some ways our system is different:

- "Files" are not user-significant, nor their rules of location or naming.

• No item (or project or document) needs to be named, unless the user wants. There are of course no limits as to name length or characters. User names are subject to Xanadu connectivity.
• Items, projects and structures may overlap in any way the user chooses.

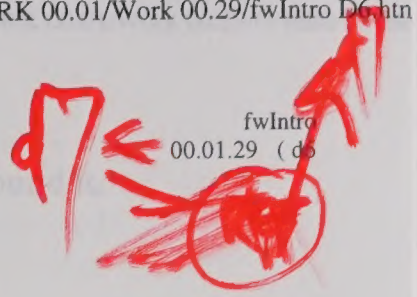
2
• Previous versions and states are all reachable.
• Backtrack time is branching and non-linear (hypertime). Group work does not require version locking because of hypertime versioning and intercalating version structure.
• The simple structure of every thinglet is easily user-accessible for change or study.

3
• The same fluid media may be used repeatedly, and you may see the different uses side by side (Xanadu transclusion).

• All your machines can share all your data uniformly across this environment.
• Hyperflobs may be published, and may yet provide an alternative to HTML, XML and their ilk.

Who could want less?

•



To Ted Nelson Home Page

FLOATING WORLD™
A NEW USER ENVIRONMENT

NO ICONS • NO METAPHORS • NO FILES • NO APPLICATIONS

For the last twenty years, as the personal computing world spilled forth, I have been increasingly indignant about all of it.

The sheer impossibility of everything I wanted to do with computers filled me with rage. The clumsy unconnected windows! The tyranny of "applications" owned by different software manufacturers! The clumsy one-step "undo"-- you can undo only one operation, and only in an application, not in the operating system! The prisons of "files" and their naming! (Conventional computer file structures have created a horrible world, and nobody realizes it, nor scarcely perceives the mess of contradictory workarounds they have engendered.)

But there is no way to get a foothold without BUYING IN-- without accepting somebody else's definition of the world. The Macintosh, Director, C++, HTML, you name it (even Unix)-- all are clumsy, and in my view, twisted tools that won't do what I think is important. The fundamental interface, designed at Xerox PARC in 1974, has not improved since that time, and has become a prison. And one of the tragedies of our time has been "word processing", especially as tangled with clumsy filing, clumsy naming, and the "clipboard"-- the computer guys thought text was simple, and therefore gave the world woefully inadequate tools.

The real problem was that we need a change at the operating system level.

Where to begin?

As Archimedes is quoted as saying, "Give me a lever long enough and I will move the world." And, of course, a fulcrum.

THE FULCRUM

I always thought that a system I designed in the 1980s, ZigZag™, had promise. It is a construct universe that lets you build any structure fairly easily, and also see it. It consists of multidimensional list structures with built-in row-and-column visualizations.

However, it took a long time to get ZigZag implemented. But now it has been prototyped and proven. In 1997, with the help of Andrew Pam of Xanadu Australia, we got a prototype running, and it has proven to be what I had imagined and visualized.

And now, with the help of Tuomas J. Lukka at Jyväskylä University in Finland, perhaps we can begin again.

STARTING OVER

I have designed a new computer universe from scratch. Prof. Lukka says he will build it.

Floating World is the name of the environment.

ZigZag is the kernel. (We will assume that you are familiar with ZigZag. If not, there is a tutorial.)

All structures are implemented in ZigZag. The user doesn't necessarily see that structure unless interested, but it's right there if you look under the hood.

Nearly everything is composed of zzcells, zzlists and permascrolls (explained below). Occasionally we also have to use conventional files, for conventional data structures which already are locked in.

ELEMENTS OF FLOATING WORLD

I. STRUCTURES

The fundamental unit is the ***thinglet***. It is not an object, an entity, a file, a table, a record, or those other nouns whose computer meanings are already committed. It is something different. Therefore a different name.

A thinglet is composed of ZZcells. It has a ***maincell***. A maincell is that cell from which the path to any of its other functional cells may be defined. The maincell has a ***handle***, free to be listed and arranged in all dimensions except d.handle, along which it attaches to the maincell.

All the internal operations of Floating World are based on thinglets, their handles and maincells, and paths between various significant cells of the thinglets.

The fundamental user-significant unit is the ***item***. An item is anything which interests a user, including: • free-floating notes • sticky notes • projects • documents • versions • graphics. Items may have many different user-definable structures, but they are fully reachable through ZigZag visualizations and construction methods.

(**TEXT**)

Text must be simple for the user, but text operations will not be simple internally, and items consisting of text may be quite complex. This is because we afford full Xanadu® connectivity to all text, even titles (see below). We relegate text to being a special case of fluid media (below).

The fundamental visualizable unit of Floating World is the ***hyperflob***, or multidimensional flying object. While many hyperflobs (or flobs for short) may appear on the screen as simple two-dimensional objects, additional dimensions may be added at any time. If you want to visualize a hyperflob, think of the old Universal Studios 3D emblem, Saturn-like, with lettering rotating around it.

All the visualizations of Floating World are based on hyperflobs and their coordinates, especially as manipulated through **edge lists**. An edge list is a zzlist of elements placed in ascending order along a single coordinate. As an element moves in the space, its coordinates change, and its ordinal position in the edge lists may also change. Tests and interpolations on these edge lists are fundamental FW operations.

We hope to provide a user programming language, **CLANG** (cell language), With Clang, we hope to make it possible for the user easily to control the multidimensional behavior of user-designed hyperflobs. It will allow both casual programming (recording keystrokes to create transposable macros) and serious programming, up to and including parallel multiprocessor programs. However, it will of course not compare in efficiency to what skilled programmers do through more conventional tools.

II. FLUID MEDIA

Cells are the structure, contents are not. Contents are either conventional files (like jpg and gif) or **fluid media**. Fluid media are digital contents which are divisible into sequentially addressable elements. Fluid media include text, audio samples, video frames, and faxes.

Fluid media are given stable addresses on **permascrolls** (append-and-read-only files). They may be linked to, and used in different places at the same time (transcluded), to the element level of granularity.

Fluid media are addressed and connected according to the Xanadu addressing model, managing both transclusions and content links. They are addressed referentially by lists (like the EDL in video editing). As in previous Xanadu designs, change is additive.

Links are based on the Xanadu xu88 model, where each link has two endsets and a type. Transclusions are discovered by address comparison.

III. ZZSTRUX + FLUID MEDIA + XUSTRUX

The combination of zzstructure, hyperflobs, fluid media and Xanadu connectivity allows virtually any presentation and media structure to be created and made radically interactive.

SYNCHRONIZATION

Fluid media are given permanent addresses, and cells are given permanent names and their contents locked for transmission. Thus Floating World contents may be exchanged with all contents permanently identified and no confusion among them.

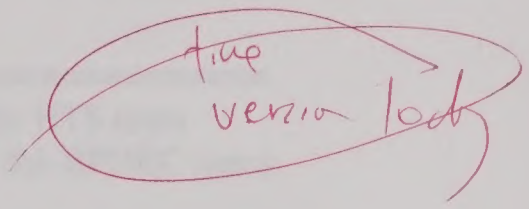
The transmission medium (a mime type) will transport clusters of cells and their associated fluid media. It will be possible to send for parts which are missing. Thus complex structures may be transmitted and synchronized between machines and users without difficulty, and without forcing destruction of previous versions (like most synchronization tools).

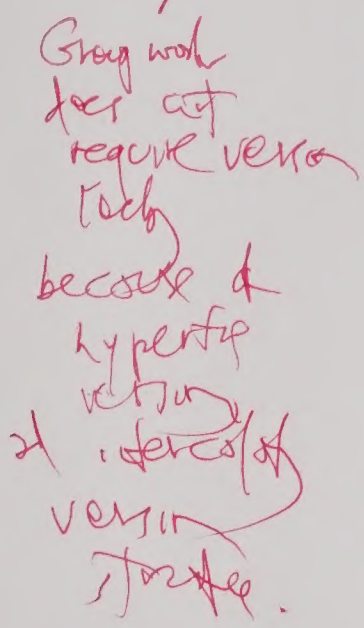
IV. DIFFERENCES FROM THE ORDINARY COMPUTER WORLD

Here is some ways our system is different:

- "Files" are not user-significant, nor their rules of location or naming.
- No item (or project or document) needs to be named, unless the user wants. There are of course no limits as to length or characters. User names are subject to Xanadu connectivity.
- Items, projects and structures may overlap in any way the user chooses.
- The same fluid media may be used repeatedly, and you may see the different uses side by side (Xanadu transclusion).
- Previous versions and states are all reachable.
- The simple structure of every thinglet is easily user-accessible for change or study.
- All your machines can share all your data uniformly across this environment.
- Hyperflobs may be published, and may yet provide an alternative to HTML, XML and their ilk.

Who could want less?





HTS dig 1/
Hek heh
12

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Sun, 06 Feb 2000 11:29:32 +0900
To: eharter@din.or.jp, marlene@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: Draft: CONGRATS TO HTS TEAM-- and the next steps
Cc: ted@xanadu.net

Edski,

What think of this? ChrzT

==========

I want to congratulate everyone on the HTS team
for doing such a fine job in meeting the JIPDEC specs
and doing impressive demos.

Now, of course, the real work begins. I am sure you
all recognize that we have not fulfilled the real objective,
which is to create a robust, distributable server that
people around the world can and will use.

I am enjoying learning about how projects are done in Japan.
So what are the next steps?

Best, Ted

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

• e-mail: ted@xanadu.net • world-wide fax 1/415/332-0136
• http://www.sfc.keio.ac.jp/~ted/ • http://www.xanadu.net
• Coordinates in USA Tel. 415/ 331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

Delivered-To: alias-xanadu-nel-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Sun, 06 Feb 2000 11:29:32 +0900
To: eharer@din.or.jp, marlene@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: Draft: CONGRATS TO HTS TEAM -- and the next steps
Cc: ted@xanadu.net

Ikkei,

What think of this? Chris?

I want to congratulate everyone on the HTS team
for doing such a fine job in meeting the JIPDEC specs
and doing impressive demos.

Now, of course, the real work begins. I am sure you
all recognize that we have not fulfilled the real objective,
which is to create a robust, distributable server that
people around the world can and will use.

I am enjoying learning about how projects are done in Japan.
So what are the next steps?

Best, Ted

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

• e-mail: ted@xanadu.net • world-wide fax 1(415)773-0136
• http://www.sfc.keio.ac.jp/~ted/ • http://www.xanadu.net

• Coordinates in USA Tel 415/ 331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

Ed's
version comment

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: eharter@mailhost.din.or.jp
Date: Sun, 6 Feb 2000 12:02:22 +0900
To: Ted Nelson <ted@xanadu.net>
From: Edward Harter <eharter@din.or.jp>
Subject: Re: Draft: CONGRATS TO HTS TEAM-- and the next steps
Cc: marlene@xanadu.net

Hi Ted,

>I want to congratulate everyone on the HTS team
> for doing such a fine job in meeting the JIPDEC specs
> and doing impressive demos.
>
>Now, of course, the real work begins. I am sure you
> all recognize that we have not fulfilled the real objective,
> which is to create a robust, distributable server that
> people around the world can and will use.
>
>I am enjoying learning about how projects are done in Japan.
> So what are the next steps?

My first impression that the overall thrust of the message is fine, but the second paragraph may be hard for some of our Keio friends to accept or agree with. Ie, getting the Crew Lab guys to start on the 'real work' may not be an easy task - they may not even have _any_ idea of how to do serious debugging, for all we know. Also, they probably don't, at this point, recognize that we haven't fulfilled the real objective: for some the JIPDEC objective is 'real' (oh my, doesn't bear much thinking about, does it;-)?), and they are certainly resisting looking at the obvious problems that exist.

But if we prepare the ground a little more, we may be able to make some real headway. I talked with Yuki for about two hours Wednesday evening, a nice chat, and at the end he came to the understanding that we need to do the installer _and_ work on the bugs. He even expressed some willingness to work with us on the bugs, probably after he gets back from a trip to his family in the States later this month.

My impression is that we are making slow progress, to varying degrees, with everyone we work with on the project - excepting, probably, Yoshi. Since Yoshi has been saying absolutely _nothing_ about bugs for the past two or three months, even when we ask him directly about the software and its condition, we don't know the degree, or depth, of his avoidance of the facts. Since the bugs are there, and easy for everyone else to see - and the others do, I

Delivered-To: alias-xanadu-ted-ted@xanadu.net
X-Sender: charter@mailhost.din.or.jp
Date: Sun, 6 Feb 2000 12:02:22 +0900
To: Ted Nelson <ted@xanadu.net>
From: Edward Baxter <ebaxter@din.or.jp>
Subject: Re: Draft: CONGRATS TO HTS TEAM -- and the next steps
Cc: marlene@xanadu.net

Hi Ted,

>I want to congratulate everyone on the HTS team
> for doing such a fine job in meeting the IIPDEC specs
> and doing impressive demos.
>
>Now, of course, the real work begins. I am sure you
> all recognize that we have not fulfilled the real objective,
> which is to create a robust, distributable server that
> people around the world can and will use.
>
>I am enjoying learning about how projects are done in Japan.
> So what are the next steps?

My first impression that the overall thrust of the message is fine,
but the second paragraph may be hard for some of our Kyoto friends to
accept or agree with. Ie, getting the Crew Lab guys to start on the
'real work' may not be an easy task - they may not even have _any_
idea of how to do serious debugging, for all we know. Also, they
probably don't, at this point, recognize that we haven't fulfilled
the real objective: for some the IIPDEC objective is 'real' (oh my,
doesn't bear much thinking about, does it:-)?, and they are
certainly resisting looking at the obvious problems that exist.

But if we prepare the ground a little more, we may be able to make
some real headway. I talked with Yuki for about two hours Wednesday
evening, a nice chat, and at the end he came to the understanding
that we need to do the installer _and_ work on the bugs. He even
expressed some willingness to work with us on the bugs, probably
after he gets back from a trip to his family in the States later this
month.

My impression is that we are making slow progress, to varying
degrees, with everyone we work with on the project - excepting,
probably, Yoshi. Since Yoshi has been saying absolutely _nothing_
about bugs for the past two or three months, even when we ask him
directly about the software and its condition, we don't know the
degree, or depth, of his avoidance of the facts. Since the bugs are
there, and easy for everyone else to see - and the others do, I

Printed for Ted Nelson <ted@xanadu.net>

believe, see them - if we persist a little longer we may be able to get someone - Yuki, maybe Yoshi as well - started working on them.

Anyway, as to timing I think it might be best until we finish up the JIPDEC event on the 15th and 16th. Yuki is working with Yoshi on the installer, and they all have their hands full making final preparations. Once that is out of the way, and we hear from Naemura-sensei that Iihashi-san and Science House are _finally_ done with those humongous docs, we can start pushing with as much persistence and determination as we wish.

Cheers,

edski

To: Tuomas J. Lukka
From: Ted Nelson <ted@xanadu.net>
Subject: :gzz,fw: Plz approve again (d7 of gzz-annoz
Cc: xted, mj, andrew, ed
Bcc:
Attached: C:\WINDOWS\DESKTOP\FlamingXpurpOnBlack-lo.JPG; M:\WORK 00.01\Work 00.29\fwIntro 00.29\fwIntro D6.html;

Subject: GZIGZAG AND FLOATING WORLD UNDERWAY IN FINLAND

==========

Hi everybody--

This announcement has been held back until we had something to show, and until I got the domain names floating-world.net and -.org.

Contents:

- Open source ZigZag™ (Gzigzag)
- Licenses (lgpl and transpub)
- FLOATING WORLD™, my new universe
- Introducing Tuomas J. Lukka
- University credit for working on Gzigzag
- About Jyväskylä University
- How to pronounce Jyväskylä

==========

- Open source ZigZag™ (Gzigzag)

An open source version of the ZigZag system, called Gzigzag, is being developed in Finland, at Jyväskylä University, under the direction of Tuomas J. Lukka. I'm the residual custodian in the event of some change. (The name "Gzigzag" is by permission, as a variant of my ZigZag trademark. See cute logo, enclosed, and, if you like, get download fixings at http://xanadu.com/zigzag for a boot floppy of the prototype that I did earlier with Andrew Pam. The floppy, if you choose to accept it, loads its own Linux onto a standard PC and lets you play with the ZZ space and structure. You can also extend it in Perl. There is also a ZigZag mailing list. Sorry, no t-shirts or mugs yet.)

The new version is more heavy-duty, Development is in Java. The current preliminary version of Gzigzag has been tested under Linux and Windows.

Current extensions in this version (beyond the prototype) include
- new rasters and visualizations, including animation
- cellscrolls retaining the entire history of a space;

and under short-term development are
- keybindings stored in the space itself
- a simple networking protocol.

The Gzigzag code and developer tools may be found at sourceforge.net, which provides tools and disk space to open source projects. (You'll also need CVS, which you'll need to find elsewhere.)

• Licenses (lgpl and transpub)

The open source license is currently lgpl, but will later be a dual license with some form of transpublishing as the other permission.

• FLOATING WORLD™, my new universe

Floating World is my new overall design for a new computer world very different from today's (see brief HTML enclosure). It will be built outward from the Gzigzag kernel, under the same open source arrangement.

I have done a very rough but detailed writeup of the Floating World design, not yet for general distribution. A rewrite will be at floating-world.net Real Soon Now. However, anyone intensely interested can get the URLs from me for the present rough working papers (about 70 pages if you print them out). No casuals plz.

• Introducing Tuomas J. Lukka <lukka@iki.fi>

Tuomas is 25 and a busy man. He got his Ph.D. at the age of 20 and has just gotten back from three years as a Harvard Junior Fellow, where he served as Chocolate Steward to the Society of Fellows and changed his field from physical chemistry to computer science.

He plays serious Go and various musical instruments, and has a two-month old daughter.

The Gzigzag project is currently his principal work, as he has a light teaching schedule.

• University credit for working on Gzigzag

This project has serious support from the university. Jyväskylä will give credit to university students from anywhere in the world who want to come to Finland to work on this project. You do not have to know Finnish; English and Java are sufficient. However, the work is taken seriously and you can't expect to coast through.

There is no tuition cost (Finland!) and living costs are roughly around U.S. level, with cafeteria meals (good ones!) somewhat subsidized.

You will of course have to make arrangements with your home university to make sure the credits will be accepted. But this is a high-class institution and should have its credits honored anywhere.

Jyväskylä University has spring and fall semesters and a one-month summer session; any of these time periods should be acceptable if you want to come pitch in and get credit.

Contact Tuomas for details.

• About Jyväskylä University

Jyväskylä University is a sparkling campus about 3 1/2 hours north of Helsinki by very nice train. "It may seem far away, but only if you're coming from somwhere else !-)" The emphasis is technical, and there is a big new research building for Nokia (the cell phone company, recently the biggest company in Europe) next door.

• How to pronounce Jyväskylä

For English speakers: say "you-vascular", without the final r. That seems to be 90% accurate.

Best, Ted

To: "Tuomas J. Lukka" <lukka@iki.fi>
From: Ted Nelson <ted@xanadu.net>
Subject: :fw: (d1) Design theory & ideas Re: ZZ, Xanadu, Dominica project
Cc: xted, mj, andrew, ed
Bcc:
Attached:

Hi Tuomas--

[from another email]
>Good. If he can spec the audio clip (or not "clip" but "window") stuff
>soon, I can maybe show that as well at Nokia.

[from still another email]
>The *MOST* important questions I would really like to have answers quick
>(everything else can wait longer) is:
>
>- how to link the sound stream and cells:
> - I-stream spans in cells?
> - how to show where the clip is going if there are several links?
> or do we play cells and skip to next cell on some dimension to play
> that next?
> - how to link along cells with overlapping I-spans, i.e. content
> links?
> (the same will probably work for text streams and a text window)
>
>- how to do synchronization / notification. This is a really thorny issue
> still and is making building the prototype hard.
>
>- in text streams or content lists, how to store the formatting info,
> ie. paragraph change, font change? In an interview or so, should something
> special be done with the names of the people speaking?

Okay, this is a lot in a hurry.

Let's see if I understand what is needed. You want--

- to have easily selectable audio
- to be able to associate the audio with text
- to embed this (somehow) in a greater structure

At 02:44 PM 1/26/00 +0200, you wrote:
>Ok, more information about the proposed project: this has potential.
>
>As I mentioned, a cultural anthropologist friend of mine (Tiina

To: "Thomas J. Lukka" <dukka@hi.fi>
From: Ted Nelson <ted@xanadu.net>
Subject: Fw: (xEwt)Design theory & ideas Re: ZZ, Xanadu, Dominica project
Cc: ted, mj, andrew, jd
Bcc:
Attached:

Hi Thomas--

[from another email]
>Good. If he can spec the audio clip (or not "clip" but "window") stuff
>soon, I can maybe show that as well at Nokia.

[from still another email]
>The *MOST* important questions I would really like to have answers quick
>(everything else can wait longer) is:
>
>- how to link the sound stream and cells:
> - I-stream spans in cells?
> - how to show where the clip is going if there are several links?
> or do we play cells and skip to next cell on some dimension to play
> that next?
> - how to link along cells with overlapping I-spans, i.e. content
> links?
> (the same will probably work for text streams and a text window)
>
>- how to do synchronization / notification. This is a really thorny issue
> still and is making building the prototype hard.
>
>- in text streams or content lists, how to store the formatting info,
> ie. paragraph change, font change? In an interview or so, should something
> special be done with the names of the people speaking?

Okay, this is a lot in a hurry.

Let's see if I understand what is needed. You want--

* to have easily selectable audio
* to be able to associate the audio with text
* to embed this (somehow) in a greater structure

At 02:44 PM 1/26/00 +0200, you wrote:
>Ok, more information about the proposed project: this has potential.
>
>As I mentioned, a cultural anthropologist friend of mine (Tina

>Kukkurainen, Cc'd) has 30 hours of interviews from various high-up
>government people and other informants on Dominica. She is going to
>start doing her PhD on this material this spring.
>
>My suggestion is that she use a Xanadu link structure: have all the
>sound data and transcription bidirectionally linked to the point where
>they are used in the text and also other structural things.
>
>The aim is that this is a simple and self-contained enough project
>with a person motivated to do this that we could get a working version
>pretty quickly.
>
>Now, what I'd propose is to try to make a CD out of her work as soon as
>possible, containing
> - all the GSM sound data and transcriptions
> - her Master's thesis
> - whatever is ready of her PhD thesis
> - possible other documents
> - compiled Java versions of all the necessary components
>that we can give out to people and say "Look, this content was done
>by a non-computer-expert".
>
>Now, again, the benefits of the deal for us:
>1) More exposure for the Xanadu linkage ideas: no silly soundclips but
> true clinks to the audio stream & transcription
>2) A demo that can be given to potential sponsors
>3) Possibly, some money: Finnish Academy has a "Innovative research
> environment" project - Pekka is looking into this.
>4) A real, self-contained example that can be used as a testbed for
> GZZ, Xanadu and other technologies
>5) A content person *in Finland* for testing the system.
>6) Through the exposure, possibly more humanists wanting to use this system
>
>For her:
>1) Possibly funding for PhD through the Academy project
>2) The presentation format can give her some renoun in her circles
>3) Better tools to create a good thesis
>4) A way to get the message of her PhD some exposure
>
>Possible downsides:
>- I'm sure you won't like the idea of putting it on a CD in a fixed form.
> Of course, we ultimately want it to be on the net, linkable to all.
> And we can do that, too. But having a simple demo simply runnable from
> a simple CD would be a BIG step forwards.
>- Self-contained data => wrong compromises on algorithms. This we must
> watch out for carefully. Finding all the cells in the world that
> reference a given I-stream is difficult.

>
>What we need right now to proceed:
>- the correct way to link audio into ZZ - tumblers??
>- the way to put text into ZZ - Xanadu V&I streams?
>
>As an optimization, would it be OK to have a flag bit for each cell,
>saying whether the contents is in a tumbler span or directly in the cell?
>
>Also, would having each cell contain a single tumbler span sound right?
>
> Tuomas
>
>
>
>

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Thu, 30 Dec 1999 02:18:38 +0900
To: ted@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: zx-annozmt (d1) How shall we announce it?

Hi Tuomas--

I think I asked you how we should announce the system,
but I don't recall whether I wrote a draft.

How's this? ==========

Subject: Open Source ZigZag Project in Finland-- All Welcome

Hi friends--

This is an announcement that I hope will be important.

An Open Source version of the ZigZag engine is underway in Finland, with a plan to make it the kernel for a whole new alternative computer environment-- without icons, metaphors, windows or "applications". All under Linux.

The engine is called Gzigzag (yes, under Gnome) and I'm currently lending the ZigZag trademark to it, though this may change. The environment is tentatively called FloatingWorld

The Open Source license is lgpl, with a parallel license planned under the transcopyright doctrine. Its Open Source status will not be revoked.

Developer-in-chief is Tuomas J. Lukka at Jyväskylä University. I am designer and residual custodian of the system.

Let me take this opportunity to introduce Tuomas. He is a delightful young Ph.D. in his first faculty position, and this is currently his main work.

He got his doctorate five years ago (at the age of twenty) in physical

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Thu, 30 Dec 1999 02:18:38 +0900
To: ted@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: zx-annount (d1) How shall we announce it?

Hi Tuomas--

I think I asked you how we should announce the system,
but I don't recall whether I wrote a draft.

How's this? ==========

Subject: Open Source ZigZag Project in Finland-- All Welcome

Hi friends--

This is an announcement that I hope will be important.

An Open Source version of the ZigZag engine is underway
in Finland, with a plan to make it the kernel for a whole new
alternative computer environment-- without icons, metaphors,
windows or "applications". All under Linux.

The engine is called Gzigzag (yes, under Gnome) and I'm currently
lending the ZigZag trademark to it, though this may change.
The environment is tentatively called FloatingWorld

The Open Source license is lgpl, with a parallel license planned
under the transcopyright doctrine. Its Open Source status will
not be revoked.

Developer-in-chief is Tuomas J. Lukka at Jyväskylä University.
I am designer and resident custodian of the system.

Let me take this opportunity to introduce Tuomas. He is a delightful
young Ph.D. in his first faculty position, and this is currently his main
work.
He got his doctorate five years ago (at the age of twenty) in physical

chemistry; since then he has spent three years at Harvard as a Junior Fellow, where he changed his field to computer science. He has a one-month old daughter and plays a number of musical instruments and serious Go.

Gzigzag is being written in Java, but will be extensible with cell routines in Java, Perl or C. It will have its own spatial programming language, Clang (for Cell Language).

FloatingWorld will be a user environment which is my personal proposal as an alternative to the PUI (PARC User Interface) which now dominates the computer world-- the conceptual structure and common appearance of Macintosh, Windows and Xwindows that makes people think some basic computer has been discovered.

The design of FloatingWorld has been largely completed, though the design documents are awaiting cosmetic rewrite. FloatingWorld has no windows, no icons, no metaphors, and no "applications". It implements the Xanadu(R) document model and is built on 3 1/2D visualizations. Through the ZigZag skeleton, the user may look inside to see all variables and operations of the system.

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

- e-mail: ted@xanadu.net • world-wide fax 1/415/332-0136
- http://www.sfc.keio.ac.jp/~ted/ • http://www.xanadu.net

- Coordinates in USA Tel. 415/ 331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

chemistry; since then he has spent three years at Harvard as a Junior Fellow, where he changed his field to computer science. He has a one-month old daughter and plays a number of musical instruments and serious Go.

Gzigzag is being written in Java, but will be extensible with cell routines in Java, Perl or C. It will have its own spatial programming language, Clang (for Cell Language).

FloatingWorld will be a user environment which is my personal proposal as an alternative to the PUI (PARC User Interface) which now dominates the computer world-- the conceptual structure and common appearance of Macintosh, Windows and Xwindows that makes people think some basic computer has been discovered.

The design of FloatingWorld has been largely completed, though the design documents are awaiting cosmetic rewrite. FloatingWorld has no windows, no icons, no metaphors, and no "applications". It implements the Xanadu(R) document model and is built on 3 1/2D visualizations. Through the ZigZag skeleton, the user may look inside to see all variables and operations of the system.

- 30 -

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

* e-mail: ted@xanadu.net * world-wide fax 1\415\332-0136
* http://www.sfc.keio.ac.jp/~ted/ * http://www.xanadu.net

* Coordinates in USA Tel. 415\331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

Printed for Ted Nelson <ted@xanadu.net>

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Thu, 30 Dec 1999 02:18:38 +0900
To: ted@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: zx-annozmt (d1) How shall we announce it?

Hi Tuomas--

I think I asked you how we should announce the system,
but I don't recall whether I wrote a draft.

How's this? ==========

Subject: Open Source ZigZag Project in Finland-- All Welcome

Hi friends--

This is an announcement that I hope will be important.

An Open Source version of the ZigZag engine is underway in Finland, with a plan to make it the kernel for a whole new alternative computer environment-- without icons, metaphors, windows or "applications". All under Linux.

The engine is called Gzigzag (yes, under Gnome) and I'm currently lending the ZigZag trademark to it, though this may change. The environment is tentatively called FloatingWorld

The Open Source license is lgpl, with a parallel license planned under the transcopyright doctrine. Its Open Source status will not be revoked.

Developer-in-chief is Tuomas J. Lukka at Jyväskylä University. I am designer and residual custodian of the system.

Let me take this opportunity to introduce Tuomas. He is a delightful young Ph.D. in his first faculty position, and this is currently his main work.
He got his doctorate five years ago (at the age of twenty) in physical

Delivered-To: alias-xanadu-net-ted@xanadu.net
X-Sender: ted@mail.sfc.keio.ac.jp (Unverified)
X-Mailer: QUALCOMM Windows Eudora Light Version 3.0.6 (32)
Date: Thu, 30 Dec 1999 02:18:38 +0900
To: ted@xanadu.net
From: Ted Nelson <ted@xanadu.net>
Subject: zx-announc (dl) How shall we announce it?

Hi Thomas--

I think I asked you how we should announce the system,
but I don't recall whether I wrote a draft.

How's this? ==========

Subject: Open Source ZigZag Project in Finland--All Welcome

Hi friends--

This is an announcement that I hope will be important.

An Open Source version of the ZigZag engine is underway
in Finland, with a plan to make it the kernel for a whole new
alternative computer environment-- without icons, metaphors,
windows or "applications". All under Linux.

The engine is called Gzigzag (yes, under Gnome) and I'm currently
lending the ZigZag trademark to it, though this may change.
The environment is tentatively called FloatingWorld

The Open Source license is lgpl, with a parallel license planned
under the transcopyright doctrine. Its Open Source status will
not be revoked.

Developer-in-chief is Tuomas J. Lukka at Jyväskylä University.
I am designer and residual custodian of the system.

Let me take this opportunity to introduce Tuomas. He is a delightful
young Ph.D. in his first faculty position, and this is currently his main
work.
He got his doctorate five years ago (at the age of twenty) in physical

chemistry; since then he has spent three years at Harvard as a Junior Fellow, where he changed his field to computer science. He has a one-month old daughter and plays a number of musical instruments and serious Go.

Gzigzag is being written in Java, but will be extensible with cell routines in Java, Perl or C. It will have its own spatial programming language, Clang (for Cell Language).

FloatingWorld will be a user environment which is my personal proposal as an alternative to the PUI (PARC User Interface) which now dominates the computer world-- the conceptual structure and common appearance of Macintosh, Windows and Xwindows that makes people think some basic computer has been discovered.

The design of FloatingWorld has been largely completed, though the design documents are awaiting cosmetic rewrite. FloatingWorld has no windows, no icons, no metaphors, and no "applications". It implements the Xanadu(R) document model and is built on 3 1/2D visualizations. Through the ZigZag skeleton, the user may look inside to see all variables and operations of the system.

- 30 -

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

- e-mail: ted@xanadu.net • world-wide fax 1/415/332-0136
- http://www.sfc.keio.ac.jp/~ted/ • http://www.xanadu.net

• Coordinates in USA Tel. 415/ 331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

---

chemistry; since then he has spent three years at Harvard as a Junior Fellow, where he changed his field to computer science. He has a one-month old daughter and plays a number of musical instruments and serious Go.

Gridzag is being written in Java, but will be extensible with cell routines in Java, Perl or C. It will have its own spatial programming language, Clang (for Cell Language).

FloatingWorld will be a user environment which is my personal proposal as an alternative to the PUI (PARC User Interface) which now dominates the computer world-- the conceptual structure and common appearance of Macintosh, Windows and XWindows that makes people think some basic computer has been discovered.

The design of FloatingWorld has been largely completed, though the design documents are awaiting cosmetic rewrite. FloatingWorld has no windows, no icons, no metaphors, and no "applications". It implements the Xanadu(R) document model and is built on 3 1/2D visualizations. Through the ZigZag skeleton, the user may look inside to see all variables and operations of the system.

- 30 -

Theodor Holm Nelson
Project Professor, Keio University SFC Campus, Fujisawa, Japan
Visiting Professor, University of Southampton, England

- e-mail: ted@xanadu.net • world-wide fax 1\415\332-0136
- http://www.sfc.keio.ac.jp/~ted/ • http://www.xanadu.net

- Coordinates in USA Tel. 415\ 331-4422
Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965

Printed for Ted Nelson <ted@xanadu.net>

```
opspecs.e4
# 95.8.21
# Split from opspecs.e3, which also went to Hop.e4
# Simplified from opspecs.e2.


# - - - - - NAMING CONVENTION
# Variable names shd be in Caps.

# - - - - - - - - - - OPERATION STRUCTURE
# So here's how we do an Operation.

# We place the initial cell numbers in a special
#  HOLDER TEMPLATE <tml.
# SETTING THE CELL NUMBERS INTO THE TEMPLATE.

# (Then we make whatever construction???)

# Then we replace the cell values
#  (especially the connections)
#  according to
#  the RECONSTRUCTION RULES.

# - - - - - - - - - - EXAMPLE: Hop cells in one dimension.
# Hopping a single cell,
#  the canonical arrangement of connections
#   A +  - B + - C + - D
#  becomes
#   A +  - C + - B +  -F

#  This method GENERALIZES TO HOPPING STRETCHES OF CELLS.
#   A +  - B...C + - D...E+  -F
#  becomes
#   A +  - D...E + - B...C+  -F


# - - - - - - - - - - - - - - - - - - - - For which the code is (ROUGHLY):

# - - - - - - - - - - INITIALIZATION
# set CurrentDimension whateverdimension
#  set Cell$A [cellno]
#  set Cell$B [cellno]
#  set Cell$C [cellno]
#  set Cell$D [cellno]
#  set Cell$E [cellno]
#  set Cell$F [cellno]

# So then what we want is this idea:
#  set A+ $D
#  set B- $E
#  set C+ $F
#  set D- $A
#  set E+ $B
#  set F- $C
```

From ted Fri Jul 28 16:22:02 1995
To: marlene@xanadu.net
Subject: They're talking about you...
Content-Length: 3708

----- Begin Included Message -----

From Fusso@aol.com Fri Jul 28 15:11:37 1995
Date: Fri, 28 Jul 1995 02:12:47 -0400
From: Fusso@aol.com
To: ted@xanadu.net
Subject: They're talking about you...
Content-Length: 3454

Hi Ted -

Just got the new issue of Microtimes and thought I'd pass this on from "Gulch Gossip" written by Jonathan E. (jeve@well.com)...to quote:

PULP FICTION

Wired's party line is that technology will make everything better in the future - but these interesting items have recently surfaced indicating that more than technology is needed when it comes to setting Wired's own house in order.

Gary Wolf's June story on Ted Nelson and "The Curse of Xanadu" set off a storm of protest, including threats of a libel suit, from Ted Nelson. I thought it was a pretty good story, have no idea as to how accurate it really was - but Ted produced a long list of what he claimed were inaccuracies. I would just observe that you don't have to be a Berkeley therapist to understand why Ted's feelings might have been hurt.

On the other hand, my reporters tell me that Eric Davis' story on Mark Pesce and Technopaganism in July's issue, with its ugly bulging head shot of Richard Dawkins on the cover, was at best incomplete and perhaps inaccurate. It's certainly an interesting and thought-provoking story - but I'm told that the section on last year's CyberSamhain at Joe's Digital Diner neglected to mention that for those who were there, besides the in-group of technopagans, it was a decidedly unsatisfactory experience. The prime problem being a complete lack of communication about what was going on, to the point of alienation, not because of the paganism and general weirdness (although some were apparently put off by that aspect) but because the in-crowd were determined to stay in by keeping others out. I guess calling outsiders "mundanes" is a pretty fair indication of pagan clannishness if not elitism. At least Davis, being "a longtime participant-observer in the Pagan community", got the drift and eventually explained it in print to us mundanes.

Meanwhile, rumors circulate about a serious rift in Wired's editorial department between the two camps of Executive Editor Kevin Kelly and Managing Editor John Battelle.

And two further indications that tech doesn't necessarily make right. The first line of Sylvia Paull's review of Jerry Kaplan's Startup: A Silicon

Valley Adventure got cut without anyone checking with her - a big crime in the writer's mind. She swears she'll never write for them again. To make matters worse, that was the line that the book jacket blurb used. And Fred Davis, a long-ago contributor to Wired but apparently not a fully paid one, received a check from Wired made out to Fred Davis. Naturally, he thought it was a settling of accounts and cashed the thing - only to receive a phone call asking him to reimburse the amount because the check was intended for another Fred Davis.

Now, I have to ask, "If simple bookkeeping or editorial good manners are beyond the reach of a presumably computer-savvy organization, why should we place much confidence in Wired's rah!-rah! attitude about what the ever-more complex informational technologies, so frequently covered in its pages, can do for our society?" Just asking - and I note that Evan I. Schwartz's piece on interactive television, also in the July issue, is pretty dismissive towards the whole idea.

Ted and Marlene - hope all goes well with you both. Spending lotsa time with Lauren these days, really excited about her job in Tokyo...summertime here and the living's easy, maybe too easy...

Take care -

Andy

----- End Included Message -----